

## DETAILED DESCRIPTION

---

[Detailed Description of the Invention]

[0001]

[Field of the Invention] This invention relates to the memory management unit which manages two or more cells on a memory.

It is related with the garbage collection device which performs the SUKABINJI processing and compaction processing to a cell, using a tray especially.

[0002]

[Description of the Prior Art] Conventionally, the art of dynamic memory control over the data used with application of managing reservation and opening of the field on a memory dynamically is known. How to perform clearly reservation of a field (it corresponds to data) which actually uses such dynamic memory control with the command in application, and opening of a field which became unnecessary, Although a field is clearly secured with the command in application, (without it opens clearly the field which became unnecessary) It is carried out by using either of the methods of making judge automatically that the field became unnecessary by the predetermined Management Department, and collecting, after opening these fields.

[0003] The method of the latter which collects the fields on a memory automatically is called the garbage collection.

It is applied to the thing disadvantageous [ being based on explicit opening among the applications which need many dynamic storage areas ], or impossible as a matter of fact. However, in order to mount these garbage collections [ like ]. Between application and garbage (it is treating part which performs garbage collection) collectors, The mechanism for performing detailed information exchange using a reference flag which is mentioned later is required, It can be called thing accompanied by extraordinary difficulty to change the existing application using the dynamic memory control which opens clearly the field which became unnecessary, and to incorporate the mechanism for a garbage collection.

[0004] Boehm and others has proposed the conservative TIBU garbage collector (conservative garbage collector) which does not require information exchange with application as a garbage collector incorporable into the existing application by fewer labors. (The following two articles have the more detailed description about this conservative garbage collector.) Literature 1 Hans-Juergen Boehm and Mark Weiser, Garbage collection in an uncooperative environment Software Practice and Experience Vol.18 (9), Sep. 1988, and P. 807-820 literature 2 Ono \*\*\*\* Other conservative garbage collection (conservative TIBU garbage collector method) information processing Vol.3 Apart from these, in order to perform a garbage collection at a high speed more, the technique using indirect addressing called a handle by the compaction processing (other one processing is SUKABINJI processing) which is performed as a garbage collection, and which is one of two processings exists.

[0005] Below, the outline of the processing performed by a garbage collector using drawing 27 - drawing 30 is explained first. Then, the garbage collector of the 1st

conventional example that performs information exchange with application using a reference flag as an example which mounts such a garbage collector is explained using drawing 31 - drawing 37, A reference flag and the garbage collector of the 2nd conventional example using a handle are explained using drawing 38 - drawing 42. Then, the garbage collector of the 3rd conventional example that mounts a conservative garbage collection is explained using drawing 43 - drawing 46.

[0006]First, the outline of the processing performed by a garbage collector is explained. Drawing 27 is a figure for explaining the composition of the heap area which a garbage collector uses, and drawing 28 is a figure showing the example of the contents (a value, data) stored in the cell which constitutes a heap area. Drawing 28 (a) shows the example of the contents of the cell in the heap area shown in drawing 27, and especially drawing 28 (b) shows the reference by the reference held at the cell shown in drawing 28 (a).

[0007]As shown in drawing 27, the arrangement of a cell is constituted by the heap area managed by a garbage collector, and, as for a cell, a numerical value or a reference is held. As shown in drawing 28 (a), there are two kinds such as the effective reference which shows the address of a cell, and the NULL reference which does not point to a cell of references. For example, it is for the cell of the effective reference 0 to hold the effective reference 7, and to point out the cell of the effective reference 7.

The arrow from a cell to the cell of the effective reference 7 of the effective reference 0 in drawing 28 (b) shows this.

(An arrow does not come out from a NULL reference, but this responds to a NULL reference not pointing to a cell.)

The following partial sequences comprise these cells [ like ] as a cell block. Drawing 29 is a figure showing the example of the cell block which comprises two or more cells. Drawing 29 (a) shows the example of the cell block constituted by assigning a value to the cell shown in drawing 27, and drawing 29 (b) shows the example of the reference relation between the cell blocks shown in drawing 29 (a).

[0008]As shown in drawing 29 (a), two or more cell blocks may exist in a heap area, but the same cell does not belong to two or more cell blocks simultaneously. The cell which belongs to neither of a cell block among the cells of a heap area is called an intact cell, and the whole field which consists of two or more continuous intact cells is called intact cell area. One specific cell block secured in a heap area is called a route cell block. Here, the field of the cell after the effective reference 14 is intact cell area, and the cell of the effective reference 0 is a route cell block.

[0009]When the effective reference which points out the cell of the head of the cell block B to one belonging to the cell block A of cells about the cell block A and the cell block B which are in a heap area is set up, it says, "Refer to the cell block B for the cell block A." a heap area -- a top -- n -- a piece -- a cell block -- X -- <sub>one</sub> -- X -- <sub>n</sub> -- it is -- a cell block -- X -- <sub>i</sub> (i= 1, 2 and 3, ..., n-1) -- a cell block -- X -- <sub>i</sub> -- <sub>one</sub> -- respectively -- referring to it -- \*\*\*\* -- the time -- " -- a cell block -- X -- <sub>one</sub> -- from -- a cell block -- X -- <sub>n</sub> -- it can follow -- " -- saying .

[0010]In drawing 29 (b), the route cell block is referring to the cell block 4, and the cell block 3 is referring to the cell block 2, and refer to the cell block 2 and the cell block 5 for the cell block 4. From a route cell block, the cell block 4, the cell block 2, and the cell block 5 can be followed, and the cell block 2 and the cell block 5 can be followed from the cell block 4. The cell block 2 can be followed from the cell block 3.

[0011]When a new storage area is needed, application, It changes into the state where a cell block can be assigned to the cell for the required size of the intact cell area (a garbage collector shall perform this allotment like the after-mentioned), and it can follow from a route block, and this "secures a cell block." The application can memorize and manage the data of a complicated data structure by securing a cell block, memorizing a numerical value within a cell block, or referring to other cell blocks. Application cannot be accessed to the cell block which cannot be followed from a route cell block.

[0012]If the data structure of the data which application manages changes, the reference relation between cell blocks will change and the cell block which cannot be followed from a route cell block as a result will increase. If intact cell area becomes small by these cell blocks [ like ], it becomes impossible to secure the cell block for the size which application requires.

[0013]A garbage collector performs the garbage collection which consists of SUKABINJI processing and compaction processing to the heap area of such a state, and, thereby, changes a heap area into the state where application can secure the cell block for required size. A garbage collector opens wide some of cell blocks which cannot be followed from a route cell block in SUKABINJI processing, Make into intact cell area the cell block opened wide, and in compaction processing. The reference relation between cell blocks being maintained without a cell belonging to two or more cell blocks, so that the numerical value in a cell block may not be changed, A cell block is moved in a heap area and the intact cell area divided (cell block which cannot be followed from a route cell block) is made to continue.

[0014]Drawing 30 is a figure for explaining the garbage collection to the heap area of the state which shows in drawing 29 (b). Drawing 30 (a) shows the result of the SUKABINJI processing to the heap area of the state which shows in drawing 29 (b), and drawing 30 (b) shows the result of the compaction processing to the heap area of the state which shows in drawing 30 (a). If SUKABINJI processing is performed to the heap area of the state which shows in drawing 29 (b), like drawing 30 (a), the cell block 1 and the cell block 3 which cannot be followed from a route cell block will be opened wide, and will serve as intact cell area. When compaction processing is performed to the heap area of the state which shows in this drawing 30 (a), like drawing 30 (b), The reference to the cell block 2 and the cell block 5 from a route cell block to the cell block 4 from reference and the cell block 4 is maintained. A part for the size of the cell block 1 opened wide and the cell block 2 are moved ahead (to the one where the value of an effective reference is smaller), A part for the size of the cell block 1 and the cell block 3 opened wide, The cell block 4 and the cell block 5 will be moved ahead, and the intact cell area corresponding to the cell block 1 and the cell block 3 will be unified with back (the effective reference 14 or subsequent ones) intact cell area.

[0015]These states of a heap area where a garbage collector changes with use of application like are managed, Although it changes into the state where the field where it continues for the size required [ a heap area ] from application by opening some of cell blocks wide if needed, unifying intact cell area, and considering it as one continuous field is securable, Various things are known by the mounting like point \*\*.

Below, the garbage collector of the 1st conventional example - the garbage collector of the 3rd conventional example are explained in order focusing on the feature for mounting.

[0016]Drawing 31 is a figure for explaining the composition of the heap area which the garbage collector of the 1st conventional example uses, and drawing 32 is a figure showing the example of the contents stored in the cell which constitutes a heap area, and the reference relation between these cells. A reference flag is used for the garbage collector of the 1st conventional example as shown in drawing 31.

The reference flag is matched with each cell.

When the numerical value is held at the matched cell, 0 is set up, and when the reference is held at the matched cell, 1 is set to this reference flag. (A reference flag shall be secured in a heap area as arrangement of the number of elements equal to the number of elements of the cell which is arrangement, and offset of arrangement is made to correspond to an effective reference)

Although the heap area shown in drawing 32 holds the same contents of a cell as drawing 28 (b), 1 is set to the reference flag corresponding to the cell of the effective reference 4 and the effective reference 5 with which the reference is held, for example.

0 is set to the reference flag corresponding to the cell of the effective reference 9 and the effective reference 10 with which the numerical value is held.

[0017]Drawing 33 is a figure showing the example of the cell block assigned in the heap area which the garbage collector of the 1st conventional example uses, and the reference relation between these cell blocks. Drawing 34 is a figure for explaining the garbage collection to the heap area of the state which shows in drawing 33. Drawing 34 (a) shows the result of the SUKABINJI processing to the heap area of the state which shows in drawing 33, and drawing 34 (b) shows the result of the compaction processing to the heap area of the state which shows in drawing 34 (a).

[0018]The composition of the cell block shown in drawing 33 is the same as the composition of the cell block shown in drawing 29 (b), The result of the compaction processing shown in drawing 34 (b) is the same as the result of the drawing 30 (b) compaction processing respectively as a result of the SUKABINJI processing shown in drawing 30 (a) as a result of the SUKABINJI processing shown in drawing 34 (a). Although it performs the SUKABINJI processing and compaction processing to a heap area, a reference flag being used especially for the garbage collector of the 1st conventional example, it explains the procedure of these processings below.

[0019]Drawing 35 is a flow chart which shows the procedure of the SUKABINJI processing A which the garbage collector of the 1st conventional example performs, and drawing 36 and drawing 37 are flow charts which show the procedure of the compaction processing A which the garbage collector of the 1st conventional example performs. Here, in order to show during use whether it can follow from a route cell block, when a flag is set up to each cell block and ends SUKABINJI processing, a flag shall show during use that the cell block which is 1 can be followed from a route cell block.

[0020]0 is first substituted for the SUKABINJI processing A to flag  $F_x$  during use of the cell block  $x$  ( $x=1-M$  ( $M$  is the number of a cell block)) (S101), A flag is initialized by 0 during the use to all the cell blocks, and the contents (reference) of the route cell block are pushed by the stack (S102). The leading cell of the cell block  $x$  which pop [ of the reference with which the stack was loaded ] is carried out (S103), and the reference by which pop was carried out points out is set as an object cell (S104), and 1 is substituted

for flag  $F_x$  during the use to the cell block  $x$  (S105).

[0021]Then, it is judged whether the reference flag of an object cell is 1 (S106), If the reference flag of an object cell is 1 (it is Yes at S106), The reference which is the contents of the object cell is pushed by the stack (S107), processing is advanced to S108, and if the reference flag of an object cell is not 1 (it is No at S106), processing will be advanced to S108, without performing processing by S107.

[0022]It is judged whether there is any cell which follows an object cell within the cell block same at S108, If there is no cell which follows (it is No at S108), the cell which will follow if there is a cell which follows (it is Yes at S108) is used as an object cell, and the processing from S(S109) 106 is repeated, and it will continue, and it will be judged whether a stack is empty (S110). If a stack is not empty (it is No at S110), the processing from S103 will be repeated, and this processing will be ended if a stack is empty (it is Yes at S110).

[0023]According to the SUKABINJI processing A with the garbage collector of these 1st conventional example [ like ]. A reference flag being used and whether the contents of the cell are references being judged. The cell block which can be followed from a route cell block will be specified, a flag will be set to 1 during use of these specified cell blocks, and cell blocks other than these will be wide opened as intact cell area.

[0024]By the compaction processing A following such SUKABINJI processing A, 1 is first substituted for  $x$  for specifying a cell block, and 0 is substituted for the reference value  $W$  for computing  $R_x$  which shows the reference value after a compaction to the cell block  $x$  (S201). It is judged whether a flag is 1 during use of the cell block  $x$  (S202), If a flag is 1 during this use (it is Yes at S202), the reference value  $W$  will be assigned to reference value  $R_x$  to the cell block  $x$  (S203), Reference value  $R_x$  of the leading cell of the cell block  $x$  after a compaction is become final and conclusive, cell several  $S_x$  of the cell block  $x$  is added to the reference value  $W$  (S204), and processing is advanced to S205. If a flag is not 1 during use (it is No at S202), processing will be advanced to S205, without performing processing by S203 and S204.

[0025]In S205, it is judged whether  $x$  is equal to  $M$ , if  $x$  is not equal to  $M$  (it is No at S205), 1 is added to  $x$ , the processing from S(S206) 202 is repeated, and if  $x$  is equal to  $M$  (it is Yes at S205), processing will be advanced to S207. (By processing by these [ S201-S206 ], the reference value after a compaction will be become final and conclusive to all the cell blocks in a heap area, a flag being used during the use set up by the SUKABINJI processing A.)

In S207, 1 is substituted for  $x$  and it is judged whether flag  $F_x$  is 1 during use of the cell block  $x$  (S208). If flag  $F_x$  is 1 during use (it is Yes at S208), 1 will be substituted for  $y$  (S209) and it will be judged whether flag  $F_y$  is 1 during use of the cell block  $y$  (S210).

[0026]If flag  $F_y$  is 1 during use of the cell block  $y$  (it is Yes at S210), the leading cell of the cell block  $y$  will be set as an object cell (S211), When the reference flag of an object cell is 1 and the contents of the object cell are the references to the cell block  $x$  (it is Yes at Yes and S213 in S212), Reference value  $R_x$  is substituted for an object cell (S214), processing is advanced to S215, and processing is advanced to S215, without performing processing by S214, when it is except these (it is No at No or S213 in S212). Processing is advanced to S215, without performing processing by S211-S214, if flag  $F_y$  is not 1 during use of the cell book  $y$  (it is No at S210).

[0027]It is judged whether there is any cell which follows an object cell within the cell

block same at S215, The cell which will follow if there is a cell which follows (it is Yes at S215) is used as an object cell, the processing from S(S216) 211 is repeated, and if there is no cell which follows (it is No at S215), it will be judged whether y is equal to M (S217). If y is not equal to M (it is No at S217), 1 is added to y, the processing from S(S218) 210 is repeated, and if y is equal to M (it is Yes at S217), processing will be advanced to S219.

[0028]If flag  $F_x$  is not 1 during use (it is No at S208), processing will be advanced to S219, without performing processing by S209-S218. In S219, it is judged whether x is equal to M, if x is not equal to M (it is No at S219), 1 is added to x, the processing from S(S220) 208 is repeated, and if x is equal to M (it is Yes at S219), processing will be advanced to S221. (By processing by these [ S207-S220 ], a reference flag being used, the reference value of the cell contained in the cell block whose flag is 1 during use so that reference relation may be maintained before and behind a compaction) It will be updated by the reference value after the previously fixed compaction. The cell block x in use [ in a heap area ] is searched, and renewal of the above-mentioned reference value is performed especially here by assigning the reference value after a compaction to the cell in the cell block y which has pointed out the cell block x.

In S221, 1 is substituted for x and it is judged whether flag  $F_x$  is 1 during use of the cell block x (S222). It is copied from the cell which reference value  $R_x$  which the contents of the cell block x carry out like the point, and by which they were computed when flag  $F_x$  was 1 during this use (it is Yes at S222) shows (S223), and processing is advanced to S224, Processing is advanced to S224, without performing processing by S223, if flag  $F_x$  is not 1 during use (it is No at S222).

[0029]It is judged whether x is equal to M at S224, if x is not equal to M (it is No at S224), 1 is added to x and the processing from S(S225) 222 is repeated, If x is equal to M (it is Yes at S224), henceforth will be made into an intact cell block from the reference value W (S226), and this processing will be ended. (Based on the reference value after the compaction for every previously fixed cell block, a cell block will actually be moved by processing by these [ S221-S226 ].)

Like these, in the compaction processing A. First, a flag being used during the use set up by the SUKABINJI processing A. The reference value after a compaction being become final and conclusive (processing by S201-S206), and a reference flag being used to all the cell blocks in a heap area. The reference value of the cell by which a flag is contained in the cell block which is 1 during use so that the reference relation between cell blocks may be maintained before and behind a compaction, It is updated by the reference value after the previously fixed compaction (processing by S207-S220), and a cell block is actually moved after that based on the reference value after a compaction (processing by S221-S226).

[0030]As mentioned above, the garbage collector of the 1st conventional example performs the SUKABINJI processing A (drawing 35) and the compaction processing A (drawing 36, 37) to the heap area which uses a reference flag. Drawing 38 is a figure for explaining the composition of the heap area which the garbage collector of the 2nd conventional example uses, and drawing 39 is a figure showing the example of the contents stored in the contents stored in the cell which constitutes a heap area, and a tray, and the reference relation between these cells.

[0031]As shown in drawing 38, a reference flag and a handle are used for the garbage

collector of the 2nd conventional example. Each cell holds a numerical value or a handle, and is specified by the effective reference, and each tray holds a reference and is specified by an effective handle. As shown in drawing 39, in a reference. There are two kinds such as the effective reference which shows the address of a cell, and the NULL reference which does not point to a cell, and there are two kinds such as the effective handle which shows the address of the tray holding an effective reference, and the NULL handle which does not point to a tray of handles.

[0032]For example, it is for the cell of the effective reference 12 to hold the effective handle 1, and to point out the tray of the effective handle 1.

It is for the tray of the effective handle 1 to hold the effective reference 4, and to point out the cell of the effective reference 4.

The arrow from the cell of the effective reference 12 in drawing 39 to the tray of the effective handle 1 and the arrow from the tray of the effective handle 1 to the cell of the effective reference 4 show these.

[0033]Drawing 40 is a figure showing the example of the cell block assigned in the heap area which the garbage collector of the 2nd conventional example uses, and the reference relation between these cell blocks. The tray which holds a NULL reference among trays in drawing 40 is called an intact tray, and the whole field which consists of two or more continuous intact trays is called intact tray area.

[0034]When the effective handle which points out the tray which holds the effective reference which points out the cell of the head of the cell block B about the cell block A and the cell block B in a heap area is held at the cell contained in the cell block A, It says, "Refer to the cell block B for the cell block A." When  $n$  cell block  $X_1 - X_n$  are on a heap area and cell block  $X_i$  ( $i=1, 2$  and  $3, \dots, n-1$ ) is referring to cell block  $X_{i+1}$  like point \*\*, respectively, It says, "Cell block  $X_{[ \text{from cell block } X_1 ] n}$  can be followed."

[0035]Here, the route cell block is referring to the cell block 4 via the tray of the effective handle 3, The cell block 3 is referring to the cell block 2 via the tray of the effective handle 1, and refer to the cell block 5 for the cell block 4 via the tray of the effective handle 4 with reference to the cell block 2 via the tray of the effective handle 1. From a route cell block, the cell block 4, the cell block 2, and the cell block 5 can be followed, and the cell block 2 and the cell block 5 can be followed from the cell block 4.

[0036]When a new storage area is needed, application, Assign a cell block to the cell for the required size of the intact cell area, and one tray in intact tray area is received, The effective reference corresponding to the leading cell of the assigned cell block is set up, it changes into the state where it can follow from a route block, and this "secures a cell block."

[0037]When it becomes impossible to secure the cell block for the size which application needs with use of application in a heap area, a garbage collector, By performing SUKABINJI processing and compaction processing, a heap area is changed into the state where application can secure the cell block for required size. Drawing 41 is a figure for explaining the garbage collection to the heap area of the state which shows in drawing 40. Drawing 41 (a) shows the result of the SUKABINJI processing to the heap area of the state which shows in drawing 40, and drawing 41 (b) shows the result of the compaction processing to the heap area of the state which shows in drawing 41 (a).

[0038]When SUKABINJI processing is performed to the heap area of the state which shows in drawing 40, like drawing 41 (a), The contents of the tray of the effective handle

0 in which either of the cell blocks is not referred to, either, and the tray of the effective handle 2 are updated by the NULL reference, and let these trays be intact trays.

According to this, the cell block 1 and the cell block 3 to which the effective reference 1 and the effective reference 6 which are the contents before renewal of the tray of the effective handle 0 and the tray of the effective handle 2 were pointing, respectively are opened wide, and let these cells be intact cell area.

[0039]When compaction processing is performed to the heap area of the state which shows in this drawing 41 (a), like drawing 41 (b), The reference to the cell block 2 and the cell block 5 from a route cell block to the cell block 4 from reference and the cell block 4 is maintained. A part for the size of the cell block 1 opened wide and the cell block 2 are moved ahead, A part for the size of the cell block 1 opened wide and the cell block 3, the cell block 4, and the cell block 5 will be moved ahead, and the intact cell area corresponding to the cell block 1 and the cell block 3 will be unified with back intact cell area.

[0040]Movement to the front of these cell blocks [ like ] is faced, the effective handle 3 which is the contents of the cell of the effective reference 0, and the effective handle 1 which is the contents of the cell of the effective reference 9 -- and, The effective handle 4 which is the contents of the cell of the effective reference 11, without being updated, The effective reference 8 which is the contents of the corresponding handle, the effective reference 3, and the effective reference 12 are updated by the effective reference 4, the effective reference 1, and the effective reference 8, respectively.

[0041]In particular, although the compaction processing B with the garbage collector of the 2nd conventional example accelerates the compaction processing A performed with the garbage collector of the 1st conventional example using a handle (indirect addressing), it explains the procedure of these processings continuously. Drawing 42 is a flow chart which shows the procedure of the compaction processing B which the garbage collector of the 2nd conventional example performs. Here in advance of execution of this compaction processing B, by the SUKABINJI processing A which the garbage collector of the 1st conventional example performs, and same SUKABINJI processing. A reference flag being used and whether the contents of the cell are references being judged. The cell block which can be followed from a route cell block shall be specified, a flag shall be set as 1 during use of these specified cell blocks, and the contents of the tray in which either of the cell blocks is not referred to, either shall be further updated by the NULL reference.

[0042]In the compaction processing B, first by the processing of S201-S206 by the compaction processing A of the garbage collector of the 1st above-mentioned conventional example, and same processing. The reference value after a compaction is become final and conclusive to all the cell blocks in a heap area, a flag being used during the use set up by SUKABINJI processing (S301 - S306).

[0043]Then, 1 is substituted for y which specifies a tray (S307), and it is judged whether the contents of the tray y are effective references (S308). When contents  $T_y$  of the tray y was an effective reference (it is Yes at S308), after reference value  $R_x$  after a compaction was set as the tray y (S309), If contents  $T_y$  of the tray y is not an effective reference (it is No at S308), while processing by S309 will not have been performed, it is judged whether y is equal to the number N of a tray (S310). If y is not equal to N (it is No at S310), 1 is added to y, the processing from S(S311) 308 is repeated, and if y is equal to



M (it is Yes at S310), processing will be advanced to S312. (The contents of the tray will be updated by processing by these [ S307-S311 ] so that the reference relation from a tray to a cell may be maintained before and behind a compaction.)

The processing by S312-S317 is the same as the processing of S221-S226 by the compaction processing A of the garbage collector of the 1st conventional example, Based on the reference value after the compaction for every cell block previously fixed by these processings, a cell block is actually moved, and this processing is ended.

[0044]Like these, in the compaction processing B. First, all the cell blocks in a heap area are received, a flag being used during the use set up by SUKABINJI processing, The reference value after a compaction is become final and conclusive (processing of S201-S206 by the compaction processing A, and same processing of S301-S306), The contents of the tray which holds an effective reference so that the reference relation from a tray to a cell may be maintained before and behind a compaction are updated (processing of S307-S311), Then, based on the reference value after a compaction, a cell block is actually moved (processing of S221-S226 by the compaction processing A, and processing of S312 to Ssame 317).

[0045]As mentioned above, it performs management of a heap area, and compaction processing B, a handle being used for the garbage collector of the 2nd conventional example.

The compaction processing A which the garbage collector of the 1st conventional example performs will be accelerated more.

The above garbage collectors of the 1st conventional example and garbage collectors of the 2nd conventional example perform a garbage collection, all judging whether the contents of the cell are numerical values, or it is a reference using a reference flag. On the other hand, the garbage collector of the 3rd conventional example performs a conservative garbage collection, without being accompanied by the detailed information exchange with application.

[0046]Drawing 43 is a figure showing the example of the contents stored in the cell of the heap area which the garbage collector of the 3rd conventional example uses. (A reference flag and a handle are not used for the heap area which the garbage collector of the 3rd conventional example uses, and the composition applies to what is shown in drawing 27.) The garbage collector of the 3rd conventional example guesses whether the contents of the cell are numerical values, or it is a reference, and a certain cell block judges whether it is a cell block which can be followed from a route cell block, and performs a garbage collection based on these. Actually, this guess is performed so that the contents of the cell which is a reference may not be mistaken as it is a numerical value. (It permits that this garbage collector may be guessed actually accidentally [ be / the contents of the cell which is a numerical value / references ].)

It is for the cell of the effective reference 12 to hold the effective reference 4, and to point out the cell of the effective reference 4 in drawing 43.

The arrow of the polygonal line from a cell to the cell of the effective reference 4 of the effective reference 12 shows this.

In this garbage collector, it may be guessed accidentally [ be / the contents of the cell which is a numerical value actually / references ]. The arrow of the curve in drawing 43 is what shows this incorrect guess, The numerical value 4 currently held at the cell of the numerical value 14 currently held at the cell of the effective reference 3, the numerical

value 0 currently held at the cell of the effective reference 9, and the effective reference 10 is guessed accidentally [ be / each / a reference ].

[0047]Drawing 44 is a figure showing the example of the cell block assigned in the heap area which the garbage collector of the 3rd conventional example uses, and the reference relation between these cell blocks. The pointer to the following cell block shall actually be stored in the head of a cell block (here, not shown). By pinpointing the position of the head of a heap area top cell block, and judging whether the data in a certain cell is what points out the head of a cell block by following these one by one, The guess which does not mistake the above contents of the cell which is a reference actually as it is a numerical value can be performed.

[0048]Although the reference relation between the cell blocks shown in drawing 44 is the same as that of what was previously shown using drawing 29 (b), since the numerical value 6 currently held at the cell of the effective reference 4 is the same value as the effective reference 6 of the leading cell of the cell block 3, it is guessed accidentally that it is a reference. Drawing 45 is a figure for explaining the garbage collection to the heap area of the state which shows in drawing 44. Drawing 45 (a) shows the result of the SUKABINJI processing to the heap area of the state which shows in drawing 44, and drawing 45 (b) shows the result of having performed the same compaction processing as what is depended on the garbage collector of the 1st conventional example to the heap area of the state which shows in drawing 45 (a). (As shown below, the garbage collector of the 3rd conventional example performs only SUKABINJI processing.) Compaction processing cannot be performed.

If SUKABINJI processing is performed to the heap area of the state which shows in drawing 44, like drawing 45 (a), the cell block 1 which cannot be followed from a route cell block will be opened wide, and let it be intact cell area. The cell block 2 which can be followed from a route cell block, the cell block 4, and the cell block 5 have become [ being left behind with as, without being opened wide, and ].

The numerical value 6 currently held at the cell of the effective reference 4 in the cell block 2 is accidentally conjectured to be a reference, and since it judges accidentally that the cell block 2 refers to the cell block 3, the cell block 3 has become [ being left behind with as, without being opened wide, and ].

[0049]If the compaction processing A by the garbage collector of the 1st conventional example of point \*\* (refer to drawing 36 and drawing 37) and same compaction processing shall be performed to the heap area of the state which shows in this drawing 45 (a), As shown in drawing 45 (b), the reference relation between cell blocks is maintained. A part for size, the cell block 2, the cell block 3, the cell block 4, and the cell block 5 of the cell block 1 opened wide will be moved ahead, and the intact cell area corresponding to the cell block 1 will be unified with back intact cell area.

[0050]In addition, since the value of the reference currently held in the cell is updated by this compaction processing so that reference relation may be maintained, The numerical value 6 of the effective reference 4 in front of the compaction guessed accidentally [ be / it / a reference ] will be updated by the value 4 which lengthened the moved value of the cell block 2. That is, in the garbage collector of the conventional example 3, since the contents of the cell which is a numerical value actually may be accidentally conjectured

to be a reference and the numerical value in a cell may be changed accidentally, compaction processing cannot be performed.

[0051] Although these garbage collectors of the 3rd conventional example that mounts a conservative garbage collector like perform only SUKABINJI processing, they explain the procedure of these processings below. Drawing 46 is a flow chart which shows the procedure of the SUKABINJI processing B which the garbage collector of the 3rd conventional example performs.

[0052] Although the SUKABINJI processing B performs the SUKABINJI processing A and same processing, in the SUKABINJI processing B, S106 in the SUKABINJI processing A and processing (refer to drawing 35) of S107 are transposed to processing of S406-S408. Namely, S406-S408 (about processing by the SUKABINJI processing B of those other than these.) It is judged whether according to the SUKABINJI processing A, first, the contents of the object cell were guessed (by an estimating method like point \*\*) (S406), and it was guessed that these contents were references (S407). If it is guessed that the contents of the object cell are references (it is Yes at S407), The contents of the object cell conjectured to be a reference are pushed by the stack (S408), If processing is advanced to S409 (it corresponds to S108 of the SUKABINJI processing A) and it is not guessed that the contents of the object cell are references (it is No at S407), processing is advanced to S409, without performing processing by S408. (Replacement of processing of these responds to the garbage collector of the 3rd conventional example performing the guess to a cell to the garbage collector of the 1st conventional example using the reference flag.)

According to the SUKABINJI processing A of the garbage collector of these 3rd conventional example [ like ]. Whether the contents of the cell are references being guessed, the cell block which can be followed from a route cell block will be specified, a flag will be set to 1 during use of these specified cell blocks, and cell blocks other than these will be wide opened as intact cell area.

[0053] As mentioned above, the garbage collector of the 1st conventional example and the garbage collector of the 2nd conventional example perform the SUKABINJI processing and compaction processing to a heap area, The garbage collector of the 3rd conventional example performs only SUKABINJI processing, and it maintains the state of a heap area so that the field for size which application needs can be secured.

[0054]

[Problem(s) to be Solved by the Invention] However, since a reference flag is used in the garbage collector of the 1st conventional example, and the garbage collector of the 2nd conventional example, The operation to a reference flag is required at the time of execution of application, the load to a processor (comparing with the application which does not use a garbage collection) will be larger, and much time will be required by execution of application. In particular, in the garbage collector of the 2nd conventional example, although compaction processing is accelerated, the handle is used and it is necessary to secure [ rather than ] more fields only to a reference flag to these.

[0055] Although it does not say that a flag is operated when performing application and the load to application execution becomes small in the garbage collector of the 3rd conventional example as compared with the garbage collector of the 1st conventional example, and the garbage collector of the 2nd conventional example, It may be guessed accidentally [ be / the contents of the cell which is a numerical value actually / references

], and since SUKABINJI processing is performed based on the possible guess result of this error, a compaction cannot be mounted. So that division of intact cell area will take place in a heap area when performing application and the field for the size demanded by application can be secured smoothly, if a compaction is not mounted, The mechanism in which two or more divided intact cell area is managed will be needed separately.

[0056] This invention is made in consideration of the above problems, and is a thing. The purpose is providing the garbage collection device which can perform a high-speed compaction, making smaller load which stops small the field which is needed for execution of \*\*, and is applied to execution of application.

[0057]

[Means for Solving the Problem] A garbage collection device concerning this invention for attaining the above-mentioned purpose, Inside of two or more cell blocks which consist of a cell which is specified by the 1st sort address and holds one data of numerical values other than the 2nd sort address and the 2nd sort address, Except for a cell block which can be followed from a predetermined route cell block by repeating reference to other cell blocks via a tray which is specified by the 2nd sort address and holds the 1st sort address, So that two or more continuous cells which do not hold data in advance of SUKABINJI processing which specifies an intact cell block, and said intact cell block and SUKABINJI processing which were specified may be arranged as one continuous field, As a cell block is moved on a memory which consists of the 1st sort address and a cell and a relation of said reference is maintained before and after movement of a cell block, it is a garbage collection device which performs compaction processing which updates the 1st sort address currently held at a tray.

[0058] This garbage collection device whether a cell contained in the 1st cell block holds the 2nd sort address. It is surmised that it does not mistake that the 2nd sort address is not held when the 2nd sort address is held, By presuming that the 2nd cell block containing a cell corresponding to the 1st sort address which a tray directed by a cell conjectured to hold the 2nd sort address holds has reference from the 1st cell block, It is characterized by specifying an intact cell block in said SUKABINJI processing.

[0059]

[Embodiment of the Invention] Hereafter, the garbage collector which is one of the examples in an embodiment of the invention is explained, referring to drawings. Drawing 1 is a figure for explaining the composition of the heap area which the garbage collector which is one of the examples of this invention uses, and drawing 2 is a figure showing the example of the contents stored in the cell which constitutes a heap area, and the reference relation between these cells.

[0060] As shown in drawing 1, the arrangement of a cell and the arrangement of a tray are constituted by the heap area managed by this garbage collector. Each cell specified by the effective reference holds a numerical value or a handle, each tray specified by an effective handle holds a reference, and it manages the state of a cell, the effective reference stored in a tray being used for this garbage collector.

[0061] In a reference, as shown in drawing 2 (moreover it is the same as that of the garbage collector of a conventional example), There are an effective reference which shows the address of a cell, and a NULL reference which does not point to a cell, and handles include the effective handle which shows the address of the tray holding an

effective reference, and the NULL handle which does not point to a tray. Here the effective handle 2 currently held at the cell of the effective reference 0, and the cell of the effective reference 4, Having pointed out the tray of the effective handle 2 is shown and it is shown that the effective handle 3 currently held at the cell of the effective reference 7 has pointed out the tray of the effective handle 3. It is shown that the effective handle 4 which similarly is held at the effective handle 1 currently held at the cell of the effective reference 12, the effective handle 0 currently held at the cell of the effective reference 15, and the cell of the effective reference 17 has pointed out the corresponding tray, respectively. Such reference relation is expressed by the arrow of the polygonal line from a cell to a tray on the figure.

[0062]In drawing 2. The effective reference 3 currently held at the tray of the effective handle 0, the effective reference 4 currently held at the tray of the effective handle 1, the effective reference 7 currently held at the tray of the effective handle 2, the effective reference 8 currently held at the tray of the effective reference 3. And it is expressed by the arrow of the polygonal line from a tray to a cell that the effective reference 14 currently held at the tray of the effective handle 4 has pointed out the cell corresponding, respectively.

[0063]Especially this garbage collector guesses whether the contents of the cell are numerical values, or it is a handle, It is presumed whether a certain cell block is a cell block which can be followed from a route cell block, A compaction is performed to the heap area which performed SUKABINJI (opening of a tray) based on these and where SUKABINJI was performed, and actually, the above-mentioned guess is performed so that the contents of the cell which is a handle may not be mistaken as it is a numerical value. (It permits that this garbage collector may be guessed actually accidentally [ be / the contents of the cell which is a numerical value / handles ].) The inside of drawing 2, The arrow of the curve from the cell of the effective reference 9 to the tray of the effective handle 0, and the arrow of the curve from the cell of the effective reference 10 to the tray of the effective handle 4, What is guessed accidentally [ be / the contents of the cell which is a numerical value actually / handles ] by the above guesses is shown.

[0064]It actually assumes that a processor (CPU) can treat two or more data memory space, For example, one data memory space with 32 bit-address space, It shall be occupied as a data segment and an extra segment, respectively, a cell (data size: 32 bits) shall be assigned to a data segment, and a tray shall be assigned to an extra segment. In this case, the tray in the extra segment which will be specified with the value which a cell holds can be inspected, and it can be guessed whether the value which a cell holds is an effective handle, or it is a numerical value according to whether this tray holds the effective reference.

[0065]The range of an address corresponding to a cell (for example, 3000-12999) and the range of the address corresponding to a tray (0-2999) shall be changed supposing the ability of a processor to treat one data memory space. In this case. [ whether the value which a cell holds is in the range of the address corresponding to a tray, and ] And whenever it inspects whether this tray holds the effective reference, and the value which a cell holds is in the range of the address corresponding to a tray and this tray holds the effective reference, the value which a cell holds can surmise that it is an effective handle. (Here, a cell is assigned to a data segment, the range of the address corresponding to a cell is set to 0-9999, a tray is assigned to an extra segment, and the range of the address

corresponding to a tray is set to 0-2999.)

The following cell blocks comprise a cell holding the above data. Drawing 3 is a figure showing the example of the cell block assigned in the heap area which this garbage collector uses, and the reference relation between these cell blocks.

[0066](Conventional example is the same as that of a garbage collector) One specific cell block secured in a heap area is called a route cell block, and the whole field which consists of a continuous intact cell of an intact cell and plurality the cell which belongs to neither of a cell block among cells is called intact cell area. The whole field which consists of a continuous intact tray of an intact tray and plurality the tray which holds a NULL reference among trays is called intact tray area. Expression of "securing a cell block" etc. "which can follow cell block  $X_{[ \text{from cell block } X_1 ] n}$ " "with which it refers to the cell block B for the cell block A" shall be used in the same meaning as the thing in the garbage collector of the 2nd conventional example.

[0067]In drawing 3, the route cell block is referring to the cell block 4 via the tray (the effective reference 8 is held) of the effective handle 3, The cell block 3 is referring to the cell block 2 via the tray (the effective reference 3 is held) of the effective handle 1, The cell block 4 is referring to the cell block 5 via the tray (the effective reference 12 is held) of the effective handle 4 with reference to the cell block 2 via the tray of the effective handle 1, and these are shown by the arrow of the polygonal line. Although the tray of the effective handle 0 holds the effective reference 1 and refers to the cell block 1, it is referred to from neither of the cell blocks.

[0068]Although the cell of the effective reference 4 contained in the cell block 2 holds the numerical value 2, this figure 2 is accidentally conjectured to be a handle, and these are shown by the curved arrow. If it becomes impossible to secure the cell block for the size which application needs with use of application in a heap area, this garbage collector will perform SUKABINJI processing and compaction processing.

[0069]Drawing 4 is a figure for explaining the garbage collection to the heap area of the state which shows in drawing 3. Drawing 4 (a) shows the result of the SUKABINJI processing to the heap area of the state which shows in drawing 3, and drawing 4 (b) shows the result of the compaction processing to the heap area of the state which shows in drawing 4 (a). If SUKABINJI processing is performed to the heap area of the state which shows in drawing 3, the contents of the tray of the effective handle 0 in which either of the cell blocks is not referred to, either will be updated by the NULL reference from the effective reference 1 like drawing 4 (a), and let this tray be an intact tray. The cell block 1 to which an effective handle will not point is wide opened by this updating, and is made intact cell area.

[0070]The cell block 2 which can be followed from a route cell block, the cell block 3, the cell block 4, and the cell block 5, The tray of the effective handle 1 which has become [ being left behind with as, without being opened wide, and ], and is used for the reference between these cell blocks (guessed normally), The tray of the effective handle 2 accidentally guessed when used for the reference to the cell block 3 from the tray of the effective handle 3, the tray of the effective handle 4, and the cell block 2 has become [ being left behind with as, without being opened wide, and ].

[0071]When the compaction processing B by the garbage collector of the 2nd conventional example of point \*\* (refer to drawing 42) and same compaction processing are performed to the heap area of the state which shows in this drawing 4 (a), like

drawing 4 (b), The reference to the cell block 2 from a route cell block to the cell block 4 from reference, the cell block 3, and the cell block 4, And the reference to the cell block 3 from the cell block 4 to the cell block 5 guessed accidentally [ reference in addition ] from the cell block 2 is maintained. A part for size, the cell block 3 - the cell block 5, and the cell block 2 of the cell block 1 opened wide will be moved ahead, and the intact cell area corresponding to the cell block 1 will be unified with back intact cell area.

[0072]Movement to the front of these cell blocks [ like ] is faced, In the effective handle 1 and row which are the contents of the cell of the effective handle 3, the effective reference 7, and the effective reference 9 which are the contents of the cell of the effective reference 0. the effective handle 4 which is the contents of the cell of the effective reference 11 -- in addition, The numerical value 2 guessed accidentally [ be / it is the contents of the cell of the effective reference 4, and / it / an effective handle ], without being updated, The effective reference 8 which is the contents of the corresponding handle, the effective reference 3, the effective reference 12, and the effective reference 6 are updated by the effective reference 6, the effective reference 1, the effective reference 10, and the effective reference 4, respectively.

[0073]Although these garbage collectors [ like ] of these perform the compaction processing B with the garbage collector of the 2nd conventional example of point \*\*, and same compaction processing, they perform SUKABINJI processing as shown below.

Drawing 5 and drawing 6 are flow charts which show the procedure of the SUKABINJI processing which this garbage collector performs. (During use, a flag is set up to each cell block so that it may be shown like the garbage collector of the 1st conventional example, etc. whether a cell block can follow from a route cell block.)

First, 0 is substituted for this SUKABINJI processing to flag  $F_x$  during use of the cell block  $x$  ( $x=1-M$  ( $M$  is the number of a cell block)) (S1), and the reference which the handle which is the contents of the route cell block points out is pushed by the stack by it (S2).

[0074]The leading cell of the cell block  $x$  which pop [ of the reference with which the stack was loaded ] is carried out (S3), and the reference by which pop was carried out points out is set as an object cell, and 1 is substituted for flag  $F_x$  during the use to (S4) and the cell block  $x$  (S5). Then, it is judged whether the contents of the object cell were guessed (by an estimating method like point \*\*) (S6), and it was guessed that the contents of this cell were handles (S7). If it is guessed that the contents of the object cell are handles (it is Yes at S7), The reference which the handle which is the contents of the object cell conjectured to be a handle points out, If it is pushed by the stack (S8), processing is advanced to S9 and it is not guessed that the contents of the object cell are handles (it is No at S7), processing is advanced to S9, without performing processing by S8.

[0075]It is judged whether there is any cell which follows within the cell block same in S9, If there is no cell which follows (it is No at S9), the cell which will follow if there is a cell which follows (it is Yes at S9) is used as an object cell, and the processing from S(S10) 6 is repeated, and it will continue, and it will be judged whether a stack is empty (S11). If a stack is not empty (it is No at S11), the processing from S3 will be repeated, and if a stack is empty (it is Yes at S11), processing will be advanced to S12.

[0076]In S12, 1 is substituted for  $y$  and it is judged whether the contents of the tray of the handle  $y$  are effective references (S13). If the contents of the tray of the handle  $y$  are

effective references (it is Yes at S13), it will be judged whether a flag is 0 during use of the cell block x which this effective reference points out (S13). If a flag is 0 during use of the cell block x (it is Yes at S14), It is judged whether back (S15) y for which the NULL reference was substituted as contents of the tray of the handle y is equal to the number N of a tray (S16), Processing is advanced to S16, without performing processing by S15, if a flag is not 0 during use of the cell block x (it is No at S14).

[0077]If y is not equal to N (it is No at S16), 1 is added to y (S17), the processing from S13 is repeated, and this processing will be ended if y is equal to N (it is Yes at S16).

That it is these SUKABINJI processing \*\*\*\* [ like ] whether the contents of the cell are handles being guessed. The cell block which can be followed from a route cell block will be specified, a flag will be set to 1 during use of these specified cell blocks, and cell blocks other than these will be wide opened as intact cell area.

[0078]Although the contents of the cell which is a numerical value actually may be guessed accidentally [ be / it / a handle ] by the above SUKABINJI processings in this garbage collector, Since the contents of the tray which indirect addressing (handle) is used for specification of a cell, and is specified by this handle in compaction processing are updated, The contents of the cell itself will not be updated and a compaction will be performed effectively (the garbage collector of the 3rd conventional example is a difference).

[0079]The cell block which will be directed by the cell it was accidentally guessed by SUKABINJI processing that was a thing holding a handle, If the numerical value it was incorrect-guessed that it was is rewritten, it may be wide opened normally by the SUKABINJI processing performed to the next. [ which will not be integrated by intact cell area ] Below, the feature of the garbage collector of the 1st conventional example - the garbage collector of the 3rd conventional example, and this garbage collector is summarized.

[0080]

[Table 1]

	セル内容の 判定方法	スカビンジに 要する時間	コンパクション の実行	コンパクション に要する時間	アプリケーション 実行の負担
第1の 従来例	リファレンス フラグ	大きい	可能	大きい	大きい
第2の 従来例	リファレンス フラグ	大きい	可能	小さい	大きい
第3の 従来例	推測	小さい	不可能		小さい
本発明	推測	小さい	可能	小さい	小さい

[0081]actually -- a reference flag -- per [ one cell (for example, it shall comprise 4 bytes) ] -- since 1 bit is required, in order to utilize a heap area more effectively, the reference flag for eight cells is packed and used for one. When assigning a value to an application execution inner cell, in order to show whether this cell is a reference or it is a numerical value, it is necessary to set up a reference flag, and access to such a reference flag takes much time. When performing SUKABINJI processing, it must detect whether the value of a cell is a reference, or it is a numerical value, and it takes much time access to the reference flag constituted as mentioned above.



[0082]As shown in [Table 1], the garbage collector of the 1st conventional example, and the garbage collector of the 2nd conventional example, In order to use a reference flag for the judgment of the contents of a cell, to exchange information with application using a reference flag during execution of application and to refer to a reference flag on the occasion of SUKABINJI processing again, When processing by the processor which the load which the load to the processor by application execution and SUKABINJI processing take will become bigger, and has fixed throughput is assumed, These processings will take much time from (comparing with the garbage collector of the 3rd conventional example, and the garbage collector of this invention). (If the garbage collector of the 1st conventional example is compared with the garbage collector of the 2nd conventional example, since a handle is used, by the compaction processing B (refer to drawing 42) which the garbage collector of the 2nd conventional example performs) . Processing of S307-S311 will set up the reference value after updating similarly. It is what has the load simple as compared with processing of the compaction processing A (refer to drawing 36 and drawing 37) which the garbage collector of the 1st conventional example performs of S207-S220, and small which compaction processing takes. To the garbage collector of the 1st conventional example, and the garbage collector of the 2nd conventional example, the garbage collector of the 3rd conventional example, and the garbage collector of this invention, It is a conservative garbage collector and is what uses a guess for the judgment of the contents of a cell, Reference of the reference flag for the operation of a reference flag and SUKABINJI processing under above application execution is unnecessary, the load which the load of application execution and SUKABINJI processing take is small, and these processings can be performed in shorter time. (Although not mentioned above, when the garbage collector of the 2nd conventional example performs compaction processing, movement of a corresponding reference flag is followed on movement of a cell, and it can be said that the load to compaction processing is large as compared with the garbage collector of this invention.) Since especially the garbage collector of the 3rd conventional example may guess the contents of a cell accidentally by SUKABINJI processing, it cannot perform compaction processing, but. SUKABINJI processing with this garbage collector makes it possible to perform compaction processing with a conservative garbage collector.

[0083]Next, the reservation system for reserving a seat according to the input from a terminal as a more concrete example of mounting to the computer (stored program) of the garbage collector concerning this invention which has been explained above is explained. Drawing 7 is a figure showing the outline of the composition of the reservation system using the garbage collector concerning this invention, and drawing 8 is a figure for explaining the composition of the reservation data base Management Department 4 included in this reservation system.

[0084]As shown in drawing 7, this reservation system includes the terminal part A1, terminal part B-2, the terminal part C3, and the reservation data base Management Department 4. In this reservation system, a request to print out files of the seat for several of these seat minutes is required of the reservation data base Management Department 4 according to the input of the number of seats of the user of either the terminal part A1, terminal part B-2 and the terminal part C3 (1 or more). The reservation data Management Department 4 does tentative reservation of 2 sets of candidates of the seat number for several required seat minutes, and returns those candidates to the terminal part which has

required the request to print out files. The terminal part to which the request to print out files was returned chooses either of these candidates by a user's input, a selected result is reported to the reservation data base Management Department 4, and the reservation data base Management Department 4 makes the actual request to print out files of the candidate whom the user chose, and ends a request to print out files.

[0085] As the reservation data base Management Department 4 shows drawing 8, the initializing section 41 which initializes the cell used with execution of a request to print out files, the application part 42 which reserves a seat for several seat minutes which controls the whole operation, and for which a user asks, and the cell substitution part 43 which assigns a value to the cell in a cell block -- and, According to the demand of the application part 42, the garbage collector 44 which assigns a cell block and performs a garbage collection if needed on a storage area is included, The garbage collector 44 contains the compaction treating part 443 which performs the cell block quota part 441 which assigns a cell block, the SUKABINJI treating part 442 which performs the above SUKABINJI processings, and the above compaction processings.

[0086] actually -- the application part 42, the initializing section 41, the cell block quota part 441, the SUKABINJI treating part 442, the compaction treating part 443, and the terminal parts A1-C3 -- and, In the cell substitution part 43 next, a request to print out files of a seat is fulfilled, the storage area which the application part 42 requires with the garbage collector 44 being secured, and SUKABINJI processing and compaction processing being performed [ processing as shown in drawing 9 - drawing 17 is performed, ] if needed. (After explaining the procedure of processing of these, change of the cell block which assumed the input of the user from the terminal part A1 - the terminal part C3 using drawing 18 - drawing 24 is explained concretely.)

The cell block used when performing processing shown in drawing 9 - drawing 17 is a partial sequence of a cell array, shall set the length of  $x$  and the contents of a cell block to  $n$ , and shall depend the reference (address) of a leading cell on the following composition.

- $C[x]$  = head reference and  $C[x+1]$  =  $n$  of the following cell block (the length of the contents of = cell block)

- In the contents of the cell block freely used with the work and  $C[x+3]$  -  $C[x+2+n]$  = application used for a flag or a compaction during  $C[x+2]$  = use, especially here, it shall be a cell as shows a route cell block below. (The value of these is assigned to the initializing section 41 by processing as shown later.)

- When fulfilling a request to print out files with a  $C[1] = 5$  and  $C[2] = 1$  and  $C[3] = \text{NULL}$  -  $C[4] = \text{NULL}$  book reservation system the garbage collector 44, It is what performs operation which was not concerned with operation (processing by the application part 42, and processing by the terminal part A1 - the terminal part C3) of application besides assignment of a cell block being required, but became independent, According to the demand from this application, when required, SUKABINJI processing and compaction processing are performed. The garbage collector 44 processes by identifying only the composition of the above cell blocks, and the SUKABINJI treating part 442, ( $C[x+3]$  It corresponds to  $-C[x+2+n]$ ) It does not identify beforehand whether the cell which shows the contents of the cell block is a numerical value, or it is a handle, and the contents of the cell are guessed (S41 of drawing 12 as specifically shown later, and processing of S42 grade).

[0087]As mentioned above, although the garbage collector 44 does not know the contents of the cell block used with application, application uses the system cell block and candidate cell block which have a data structure as shown below in order to fulfill a request to print out files.

System cell block (the reference of the leading cell is made into  $x_0$ )

$C[x_0] - C[x_0+2] =$  - The reference to the candidate cell block of handle and  $C[x_0+4] =$  terminal part B-2 which stores the reference to above-mentioned  $C[x]$  - the candidate cell block of the value and  $C[x_0+3] =$  terminal part A1 corresponding to  $C[x+2]$  respectively. The candidate cell block to the handle terminal part t (=A, B, C) which stores the reference to the candidate cell block of the handle and  $C[x_0+5] =$  terminal part C3 to store (the reference of the leading cell is made into  $x_t$ )

-  $C[x_t] - C[x_t+2] =$  the seat number of the seat which carried out tentative reservation respectively as the value and  $C[x_t+3] =$  reservation seat number and  $C[x_t+4] - C[x_t+3 + \text{reservation seat number}] =$  candidate 1 corresponding to above-mentioned  $C[x] - C[x+2]$   
- The seat number of the seat which carried out tentative reservation as the  $C[x_t+4 + \text{reservation seat number}] - C[x_t+3 + 2 \times \text{reservation seat number}] =$  candidate 2

The notation as shown below shall be used in explanation of a flow chart among a flow chart.

\*\*C: A cell array and N: The number of a cell (size of a cell array)

-  $C[1] - C[N]$  : A cell and T: Tray arrangement and M: The number of a tray (size of tray arrangement)

-  $T[1] - T[M]$  : A tray and NULL: Numerical 0 and U: The head reference and W of an intact cell block: The work and S at the time of a compaction: Reference stack drawing 9 is a flow chart which shows the procedure of processing by the application part 42, Drawing 10 is a flow chart which shows the procedure of processing by the initializing section 41.

[0088]As shown in drawing 9, in processing of the application part 42, first, the initializing section 41 is called, the cell block on a storage area is initialized (procedure in which drawing 10 is used and shown later), and a route cell block is set up by it again (S21). Then, 3 is substituted for the required cell n [several], the cell block quota part 441 is called, the handle h obtained by processing in the cell block (based on procedure as later shown in drawing 11) quota part 441 is substituted for  $C[4]$ , and a system cell block is secured (S22).

[0089]Then, 1 is substituted for s (S23), s is substituted for k used for processing, and the terminal part s (it is  $s = 1$ , and 2 and 3 and the terminal part 1 - the terminal part 3 are made to correspond to the terminal part A1 - the terminal part C3, respectively) is called (S24). In the terminal part s, in details more by a procedure as later shown in drawing 15 and drawing 16. Processing of assignment of a cell block, substitution of the value to a cell block, etc. will be performed, the response to the demand of a request to print out files of a user being performed, and a garbage collection being performed if needed, and the tentative reservation of a seat and this request to print out files will be performed one by one. After the call of the terminal part s, 1 is added to s (S25), the processing (it is Yes at S26) from S23 or the processing (it is No at S26) from S24 is repeated according to whether the value of s is larger than three, or it is small, and the call of the terminal part A1, terminal part B-2, and the terminal part C3 is repeated successively.

[0090]Although the initializing section 41 is called in processing by the 42 above-

mentioned application partS21, As shown in drawing 10, in processing of the initializing section 41, first, 1 is substituted for x (S2101), NULL is substituted for C[x] (S2102), and it is judged whether x is equal to the number N of a cell (S2103). If x is not equal to N (it is No at S2103), 1 is added to x, the processing from S(S2104) 2102 is repeated, and if x is equal to N (it is Yes at S2103), processing will be advanced to S2105. (The processing by S2 101 - S2104 of these initializes all the cell C [1] - C [N] to NULL.)

Then, in S2105, 1 is again substituted for x, NULL is substituted for T[x] (S2106), and it is judged whether x is equal to the number M of a tray (S2107). If x is not equal to M (it is No at S2107), 1 is added to x, the processing from S(S2108) 2106 is repeated, and if x is equal to M (it is Yes at S2107), processing will be advanced to S2109. (The processing by S2105-S2108 of these initializes all the tray T [1] - T [M] to NULL.)

When 5 is substituted for C [1] and 1 is substituted for S2109 by C [2], a route cell block is set up, 5 is substituted for the head reference U of an intact cell block (S2109), the processing by the initializing section 41 is ended and the cell block quota part 441 is called by processing by the 42 above-mentioned application partS22 after this.

[0091]Drawing 11 is a flow chart which shows the procedure of processing in the cell block quota part 441, Drawing 12 is a flow chart which shows the procedure of processing by the SUKABINJI treating part 442 called from the cell block quota part 441, and drawing 13 and drawing 14 are flow charts which show the procedure of processing by the compaction treating part 443 called from the cell block quota part 441.

[0092]As shown in drawing 11, in processing of the cell block quota part 441, it is judged first whether the required cell n [ several ] is the following (N-U-3) (S2201). (that is, is reservation of a required cell possible on a storage area or not?) If n is the following (N-U-3) (it is Yes at S2201), processing will be advanced to S2205, if n is not the following (N-U-3) (it is No at S2201), (It processes in the procedure later shown in drawing 12)

The SUKABINJI processing treating part 442 and (it processes in the procedure shown in next drawing 13 and drawing 14) the compaction treating part 443 are called in order (S2202, S2203), Again, it is judged whether the required cell n [ several ] is the following (N-U-3) (S2204). Abnormal termination is carried out as what is insufficient of the cells which can be used if n is not the following (N-U-3) (it is No at S2204), and if n is the following (N-U-3) (it is Yes at S2204), processing will be advanced to S2205.

[0093]In S2205, 1 is substituted for h and it is judged whether h (updated behind) is below the number M of a tray (S2206). Abnormal termination is carried out as what is insufficient of the trays which can be used if h is not below M (it is No at S2206), and if h is below M (it is Yes at S2206), it will be judged whether T [h] is NULL (S2207). If T [h] is not NULL (it is No at S2207), 1 will be added to h (S2208), the processing from S2206 will be repeated, and if T [h] is NULL (it is Yes at S2207), processing will be advanced to S2209.

[0094]h corresponding to the tray which is not used by processing by S2205-S2208 is specified, n is substituted for T [h] by the head reference U of an intact cell block, and is substituted for C [U] by (U+n+3) and C [U+1], respectively, (n+3) is added to U (S2209), and processing in the cell block quota part 441 is completed. (These cell block quota parts [ like ] 441) The handle h which specifies the tray which secures a system cell block according to the call by the 42 application partS22 which sets the required cell n [ several ] to 3, and stores the head reference of this system cell block is outputted, The handle h which specifies the tray which stores the head reference of the candidate cell block which

secures a candidate cell block according to the call by S2411 of the terminal part A1 which makes the required cell n [ several ] (+one 2x seat) - the terminal part C3, and is newly secured is outputted.

Although the SUKABINJI treating part 442 is called in the 441 above-mentioned cell block quota partS2202, as shown in drawing 12, by processing of the SUKABINJI treating part 442, first, 1 is substituted for x (S31) and it is judged whether C[x] is NULL (S32). If C[x] is not NULL (it is No at S32), x will be updated by the value which 0 is substituted for C [x+2], and C[x] shows, the processing from S(S3 3) 32 will be repeated, and if C[x] is NULL (it is Yes at S32), processing will be advanced to S34. (Here, C [x+2] is used as a flag during use, and the processing by these S31-S33 initializes a flag to 0 during use of the cell block currently used.)

In S34, 1 is pushed by the reference stack S and it is judged whether S (updated behind) is empty (S35). If S is not empty (it is No at S35), pop [ of the reference (head reference of the cell block currently used) by which the stack is carried out to S ] will be carried out to r (S36), and it will be judged whether r is NULL (S37). If r is NULL (it is Yes at S37), the processing from S35 will be repeated, If r is not NULL (it is No at S37), it will be judged whether C [r+2] is 0 further (S38). (in order to avoid that processing not more than S35 serves as an infinite loop by the structure of a cell block)

[0095]If C [r+2] is not 0 (it is No at S38), the processing from S35 will be repeated, if C [r+2] is 0 (it is Yes at S38), a flag will be set to 1 during use of the cell block which 1 is substituted for C [r+2], and is used for it (presumed), and 0 will be substituted for x (S39). Then, it is judged whether x (updated behind) is smaller than the value of C (the size of the contents of a cell block is shown) [r+1] (S40). (were the following processings performed about all the cells contained as contents of a cell block or not?) If x is not smaller than the value of C [r+1] (it is No at S40), processing is moved to S35 (in order to set the following cell block as a target), \*\*\*\*\* [ that the value of C [r+3+x] is below or more 1M continuously if x is smaller than the value of C [r+1] (it is Yes at S40) ] (S41) -- and T[-- it is judged whether the value of C [r+3+x]] is NULL (S42). The value of C [r+3+x] is below or more 1M (it is Yes at S41), and T [C[r+3+x]] is pushed by the reference stack S only within the case where the value of T [C[r+3+x]] is not NULL (it is No at S42) (S43), and After these, 1 is added to x, the value of x is updated (S44), and the processing from S40 is repeated.

[0096](A cell block is followed one by one) If the reference stack S becomes empty (it is Yes at S35), \*\*\*\*\* [ that 1 is again substituted for x (S45), and T[x] is NULL ] (S46) -- and it is judged whether C [T[x]+2] is 1 (S47). (is a flag 1 during use of the cell block which T[x] directs or not?)

[0097]after NULL GA substitution will be carried out at T[x] (S48), if the time (it is No at S46) C [T[x]+2] of T[x] not coming out in NULL was not 1 (it is No at S47), If C [T[x]+2] is 1 when T[x] is NULL (it is Yes at S46) or (it is Yes at S47), while processing by S48 will not have been performed, 1 is added to x (S49) and it is judged whether x is below N (S50). If x is below N (it is Yes at S50), the processing from S46 will be repeated, If x is not below N (it is No at S50), the processing by the SUKABINJI treating part 442 will be ended, and the compaction treating part 443 will be called after this by processing (refer to drawing 11) in the 441 above-mentioned cell block quota partS2203.

[0098]Especially the processing by S41 and S42 of the above SUKABINJI treating parts 442, It is a thing in order to guess whether the cell to the contents of a cell block holds the

handle, or the numerical value is held, whenever the value of the tray which the value which a cell holds is a value which a handle (effective) can take, and is specified by this handle does not come out in NULL, the value which this cell holds is presumed to be a handle. According to this presumption, by S43, the reference stored in the tray which the value of the cell presumed to be a handle directs will be pushed by the reference stack S, and what the cell block specified by this reference is using will be presumed. The processing by S45-S50 of the above SUKABINJI treating parts 442 substitutes NULL for the tray which will direct the cell block which is not used, and let it be an intact tray. (A tray will be effectively reused by processing by S45-S50.)

As shown in drawing 13 (and drawing 14), in processing of the compaction treating part 443. First, 1 is substituted for the works W (the size of the cell block in use is added to the reference of the leading cell of a cell block one by one), and U (updated so that it may point to the reference of the leading cells of all the cell block in order) (S51), It is judged whether C [U+2] is 1 (S52). (is a flag 1 during use or not?) after W will be substituted for C [U+2] (the reference of the leading cell after cell block movement stored in C [U+2]) and a value (C [U+1] +3) (size of a cell block) will be applied to W, if C [U+2] was 1 (it is Yes at S52), If C [U+2] is not 1 (it is No at S52), while the value of C [U+2] and the value of W have not been changed by it, the value (the head reference of the target cell block is shown below) of C [U] will be assigned to U, U will be updated (S54), and it is judged whether U is NULL (S55). If U is not NULL (it is No at S55), the processing from S52 will be repeated, and if U is NULL (it is Yes at S55), processing will be advanced to S56. (The reference of the leading cell after cell block movement is computed by processing by S51-S55 of these, and it is stored in the predetermined work (cell corresponding to C [x+2] of each cell block) to each cell block.)

In S56, 1 is substituted for x and it is judged whether T[x] is NULL (S57). When T[x] was not NULL (it is No at S57), after the value (value of the work to each of above-mentioned cell blocks) of C [T[x]+2] was assigned to T[x], If T[x] is NULL (it is Yes at S57), while the value of T[x] will not have been changed, it is judged whether x is equal to the number M of a tray (S59). If x is not equal to M (it is No at S59), 1 is added to x (S60), the processing from S57 is repeated, and if x is equal to M (it is Yes at S59), processing will be advanced to S61. (It is updated by processing by S56-S60 of these so that all the T [1] - T [M] may direct the reference of the leading cell after cell block movement.)

In S61, 1 is substituted for U (updated so that it may point to the reference of the leading cell of a cell block in use in order), and it is judged whether the value of C [U+2] is 0 (S62). (is that the cell block is not used by C [x+2] shown or not?)

[0099]If the value of C [U+2] is not 0 (it is No at S62), (C[U+2]+C[U+1]+3) will be substituted for C [C[U+2]] (S63) (the reference of the leading cell of the following cell block (movement back) substituted for leading cell C[x] of the cell block after movement), C [U+1] The contents of -C [U+C[U+1]+2] are copied to C [C[U+2]+1] - C [C[U+2]+C[U+1]+2] (S64), and processing is advanced to S65. Processing is advanced to S65, without performing processing by S63 and S64, if the value of C [U+2] is 0 (it is Yes at S62).

[0100]In S65, C [U] is substituted and updated by U, it is judged whether U is NULL (S66), if U is not NULL (it is No at S66), the processing from S62 will be repeated, and if U is NULL (it is Yes at S66), processing will be advanced to S67. (A cell will actually be

moved by processing by S61-S66 of these.)

In S67, the value (the head reference of the cell which is not used is shown) of W is assigned to x, NULL is substituted for C[x] (S68), and it is judged whether x is equal to the number N of a cell (S69). If x is not equal to N (it is No at S69), 1 is added and updated by x (S70), if x is equal to N (it is Yes at S69), the processing by the compaction treating part 443 will be ended, and the processing (refer to drawing 11) from the 441 cell block quota part S2204 will be continued. (By processing by S67 to S69, NULL is substituted for the cell after W for a reference.)

The cell block presumed that the processing by these compaction treating parts [ like ] 443 is not [ be / it ] under use by processing by the SUKABINJI treating part 442 is unified to intact cell area.

[0101]The terminal part A1 - the terminal part C3 are called by the processing (refer to drawing 9) by the 42 application part S24 of point \*\*. Drawing 15 and drawing 16 are flow charts which show the procedure of processing in each of the terminal part A1 - the terminal part C3, and drawing 17 is a flow chart which shows the procedure of processing in the cell substitution part 43 called from the terminal part A1 - the terminal part C1.

[0102]As shown in drawing 15 (and drawing 16), in the terminal part s (respectively [ the terminal part A1 - the terminal part C3 ]), it is judged first whether a new request to print out files was demanded by the user (S2401). if a new request to print out files is required (it is Yes at S2401) -- C[-- \*\*\*\*\* [ that T [C[4]]+2+k] is NULL ] (the handle which stores the reference to the candidate cell block for the terminal part s is directed.) the cell of a system cell block -- NULL -- it is -- it being judged (S2407) and, If C [T[C[4]]+2+k] is NULL (it is Yes at S2407), processing will be advanced to S2410 as it is, If C

[T[C[4]]+2+k] is not NULL (it is No at S2407), C[-- the tentative reservation in the candidate cell block which T [C[4]]+2+k] will point out is canceled (S2408) -- C[-- after NULL is substituted for T [C[4]]+2+k] (S2409), processing is advanced to S2410.

[0103]The input of the reservation seat number by a user is received, and it is substituted for S2410 by x, (It corresponds to the contents of a cell block) The number of required cells is set to (2xx+1), the cell block quota part 441 is called (S2411), in the cell block quota part 441, processing as shown in drawing 11 is performed, and the cell block corresponding to a candidate cell block is secured. Then, it differs in an input, the cell substitution part 43 is called twice (S2412, S2413), and a value is assigned to the cell corresponding to the contents of a cell block of a system cell block and a candidate cell block (procedure shown in drawing 17 below).

[0104]Then, a reservation data base is searched based on the reservation seat number x, and tentative reservation of the two candidates is carried out, Two candidates by whom tentative reservation was done are substituted as contents of a cell block of the candidate cell block which the handle h directs (S2414), two candidates are displayed (S2415), and the processing by the terminal part s is ended. If a new request to print out files is not required (it is No at S2401), it is judged whether one of two (tentative reservation will be carried out as mentioned above) candidates was chosen by the user (S2402). If one of two candidates is not chosen (it is No at S2402), the processing by the terminal part s is ended and one of two candidates is chosen as it is (it is Yes at S2402), processing will be advanced to S2403.

[0105]The parameter which specifies either of two candidates is substituted for S2403 by x, The actual request to print out files of the candidate corresponding to the parameter x



in the candidate cell block which C [T[C[4]]+2+k] will direct will be made (S2404), NULL is substituted for C [T[C[4]]+2+k], and it changes into the state where the cell in the system block which directs the candidate cell block which ended this request to print out files is intact (S2405).

[0106]After these, it indicates that it completed this request to print out files (S2406), and it ends the processing by the terminal part s. During [ S2412 ] processing of the above-mentioned terminal part s, although the cell substitution part 43 is called in S2413, a value for the offset over the cell to which the handle h which will direct the head reference of a cell block in advance of processing assigns a value to n to substitute for a cell at m is assigned to v. \*\*\*\*\* [ that T [n] is NULL first in processing in the cell substitution part 43 as shown in drawing 17 ] (S71) -- and it is judged whether m is less than more than 0C [T[n]+1] (S72). If T [N] is NULL (it is Yes at S71), a handle will be ended as an unusual thing, and if m is not less than more than 0C [T[n]+1] (it is No at S72), offset will be ended as an unusual thing.

[0107]If T [n] is [ No) and m ] less than more than 0C [T[n]+1] not in NULL but in (S71 (it is Yes at S72), v will be substituted for C [T[n]+3+m] (S73), and processing in the cell substitution part 43 will be ended. (Substitution of the value to the cell in a system cell block is performed by processing of the terminal part s which substitutes (k-1) for m, substitutes h for v for C [4] at n, and calls the cell substitution part 43 of S2412, and) Substitution of the value to the cell in the candidate cell block corresponding to the terminal part s is performed by processing of the terminal part s which substitutes 0 for m, substitutes x for v for h at n, and calls the cell substitution part 43 of S2413.

When the cell block quota part 441 is called by processing of the above terminal parts s of S2411 and a candidate cell block is especially secured on a storage area, Processing by the SUKABINJI treating part 442 by the garbage collector 44 (refer to drawing 8) of this invention and processing by the compaction treating part 443 are performed like the time of a system cell block being secured if needed.

[0108]If operation of the user in the terminal part A1 as sets the number of 50 (it is supposed that the value of explanation small for convenience will be taken) and a tray to 4 and actually shows the number of a cell in [Table 2] below - the terminal part C3 is assumed, Processing with this reservation system shown in drawing 9 - drawing 16 will be accompanied by change of the cell on a storage area as shown in drawing 18 - drawing 23.

[0109]

[Table 2]

	ユーザからの操作
操作 I	端末部Aからの9席の予約
操作 II	端末部Bからの4席の予約
操作 III	端末部Aの予約終了
操作 IV	端末部Cからの7席の予約

[0110]Drawing 18 is a figure showing the state of the cell immediately after the operation I in [Table 2], and the end of operation II, Drawing 19 is a figure showing the state of the cell in front of the processing (refer to drawing 12) under [ S34 ] SUKABINJI processing for the operation IV after the end of operation III, Drawing 20 is a figure showing the state of the cell in front of the processing under [ S45 ] SUKABINJI processing for the



operation IV, and drawing 21 is a figure showing the state of the cell immediately after the end of SUKABINJI processing for the operation IV. Drawing 22 is a figure showing the state of the cell in front of the processing (refer to drawing 13) under [ S61 ] compaction processing for the operation IV, and drawing 23 is a figure showing the state of the cell immediately after the end of compaction processing for the operation IV.

[0111]Immediately after having required the request to print out files of 9 seats from the terminal part A1 by the operation I, and requiring a request to print out files of 4 seats from terminal part B-2 by the operation II, as shown in drawing 18, even 44 will be used among the cells of 50 on the storage area which the garbage collector 44 manages. Then, if one of the candidates is chosen from the terminal part A1 by the operation III, as shown in drawing 19, according to this, the value of the reference 8 in a system cell block will be changed into NULL (value 0) (it corresponds to processing by S2405 of drawing 15 in the terminal part A1). then, a request to print out files of 7 seats requires from the terminal part C3 by the operation IV -- having (it corresponds to processing by S2410 of drawing 15 in the terminal part C3). Since  $15 (=2 \times 7 + 1)$  required cells are not below two ( $=N-U-3=50-45-3$ ), it is judged with it being necessary to perform SUKABINJI processing and compaction processing (it corresponds to No by processing by S2201 of drawing 11).

[0112]According to processing (refer to drawing 12) until just before under [ S34 ] SUKABINJI processing, first, A flag is initialized by 0 during use of a route cell block, a system cell block, the candidate cell block to the terminal A1, and the candidate cell block to terminal B-2 (here, since the NULL value which initializes a cell is set to 0, the value of a flag is not changed during use).

[0113]the value (the value which route cell block inner cell C [4] holds.) of the cell in which the garbage collector 44 (refer to drawing 8) holds the contents of a cell block The value which system cell block inner cell C [8] - a cell [10] hold, respectively, The value which candidate cell block inner cell C [14] to the terminal part A1 - cell C [32] hold, respectively, . [ whether the value which candidate cell block inner cell C [36] to terminal part B-2 - cell C [44] hold is a thing corresponding to a numerical value, and ] It cannot be become final and conclusive beforehand whether it is a thing corresponding to a handle, and it is guessed by the processing by S41, and processing by S42 whether the value of a cell is a thing corresponding to a numerical value or it is a thing corresponding to a handle. (Since the reference relation between cell blocks is specified, such a guess is required.)

The value 3 which the cell C [9] in the value 1 which the cell C [4] in a route cell block holds, and a system cell block actually holds, Since each of values 5 and 33 which the tray which is in the range which an effective handle can take and is directed by this effective handle holds are not NULL(s) (value 0), These cell C [4] and the cell C [9] are presumed to all hold the handle, (1 is substituted for a flag during the use corresponding to each cell block) What the system cell block directed with the value in the tray directed by the handle of these and the candidate cell block to terminal part B-2 are using will be presumed.

[0114]In on the other hand, the value 0 and row which cell [ in a system cell block ] C [8] and the cell C [10] hold. The value 9 which the cell C [14] in the candidate cell block to the terminal part A1 holds, The value 4 which is not in the range which an effective handle can take, and the cell C [36] in the candidate cell block to terminal part B-2 holds,

Although it is in the range which an effective handle can take, 0 is held at the tray directed by this, and these cell C [8], cell C [10], cell C [14], and the cell C [36] are conjectured to all hold the numerical value. (The same guess is performed also to cell C [37] which is the contents of a cell block of the candidate cell block to cell C [15] - cell C [32], and the terminal part B1 which show the contents of a cell block of the candidate cell block to the terminal part A1 - cell C [44].)

According to the guess about the value of the cell corresponding to these contents [ like ] of a cell block, as shown in drawing 20, 1 will be substituted for a flag (cell C [3], cell C [7], and cell C [35]) during use of each (it was presumed that it was directed on the tray) cell block presumed to be under use. Then, the value of tray T [2] which directs the cell block conjectured not to be used by the processing after during [ S45 ] SUKABINJI processing is set to NULL, and a cell will be in the state where it is shown in drawing 21. [0115]By compaction processing which the reference relation of a cell block is presumed by the SUKABINJI processing accompanied by these the guesses of those that the value of a cell is a numerical value like, or are handles, and is performed next. A cell will be in the state which shows in drawing 23 through the state which shows in drawing 22, and will be in the state where the field for memorizing the seat number (and the number of seats) to the seat for 7 seats demanded on the occasion of the operation IV is securable. [0116]As mentioned above, whether this reservation system is direct in operation of application called a request to print out files of a seat and the changing portion (the application part 42, the initializing section 41, the cell substitution part 43 and the terminal part A1 - the terminal part C3), The portion besides the garbage collector 44 (except [ that reservation of a cell block is directed ]) operates independently including the garbage collector 44 which performs reservation and opening of a storage area according to the demand from application. Since especially the garbage collector 44 is what performs SUKABINJI processing, guessing whether the contents of a cell are numerical values, or it is a handle (without using a reference flag etc.), Reservation of the field for a reference flag etc. can be made unnecessary, and it can prevent applying excessive load to a processor when performing application, and a high-speed compaction can be performed effectively.

[0117]Then, the modification of these garbage collectors [ like ] 44 is explained. In the garbage collector of this modification, a disable flag is set up to each tray, and assignment of a cell block is performed, these disable flags being used. (The composition of the garbage collector of this modification, operation, etc. shall apply to the garbage collector 44 except for what is shown below)

Drawing 24 is a flow chart which shows the processing (it corresponds to the processing shown in drawing 10) by the initializing section performed in advance of execution of the garbage collector of this modification, Drawing 25 is a flow chart which shows processing (it corresponds to the processing shown in drawing 11) in the cell block quota part of the garbage collector of this modification, and drawing 26 is a flow chart which shows processing (it corresponds to the processing shown in drawing 17) in the cell substitution part of the garbage collector of this modification.

[0118]In the initializing section of a reservation system to the garbage collector of this modification, tray T[x] follows on being initialized by NULL and disable flag D[x] corresponding to this tray is initialized by zero so that it may be shown as processing of drawing 24 of S3106. In the cell substitution part of the garbage collector of this

modification. It is made to be shown as processing of drawing 26 of S3074 and S3075, One is substituted for corresponding disable flag D[x] in order to forbid use of this tray, when the value assigned to the cell used is in the range of the value corresponding to a handle and the tray corresponding to that handle holds NULL.

[0119]In a cell block quota part, the handle h corresponding to the cell block to which the disable flag which holds NULL and corresponds is newly assigned to the tray which is 0 will be specified so that it may be shown as processing of drawing 25 of S3207. In a cell block quota part, if h is not below the number M of a tray so that it may be shown as processing of S3210-S3215 (it is No at S3206), all the disable flag will be set to 0, prohibition of use of a tray will be canceled, and processing from S3205 will be performed again.

[0120]First, 0 is substituted for the work flag f, 1 is substituted for i (S3210), and it is judged whether disable flag D[i] corresponding to tray T[i] is 1 (S3211). If D[i] was 1 (it is Yes at S3211), after 1 will be substituted for f and zero will be substituted for D[i] (S3212), and if D[i] is not 1 (it is No at S3211), processing by S3212 will be performed [ as for which things are not ]. It is judged whether i is equal to the number M of a tray (S3213). If i is equal to M (it is Yes at S3213), if i is not equal to M (it is No at S3213), 1 is added to i and the processing from S3211 is repeated, and it will continue, and it will be judged whether f is 1 (S3215).

[0121]If f is not 1 (it is No at S3215), will carry out abnormal termination as what is insufficient of trays, if f is 1 (it is Yes at S3215), it will be advanced to S3205 by processing, and by processing after this. The handle h corresponding to the cell block to which the disable flag which holds NULL and corresponds is newly assigned to the tray changed into 0 will be specified.

[0122]Namely, by processing in the above cell substitution parts, make into a disable the tray corresponding to the value of a cell it will be guessed by next SUKABINJI processing that is a numerical value, and by processing in the above cell block quota parts. The contents of a cell in a SUKABINJI treating part can be made to guess more appropriately, when the disable flag which holds NULL and corresponds specifies the handle h corresponding to the cell block newly assigned only to the tray which is 0.

[0123]In details, more in processing by the SUKABINJI treating part of the garbage collector 44. The value in a cell is in the range of the value corresponding to a handle, and when it considers that the value in this cell is a handle, a cell is conjectured to hold the handle whenever the tray specified by this handle does not hold NULL. By setting the disable flag corresponding to this tray to 1, when the numerical value in the range of the value corresponding to a handle is substituted for a cell and the tray which is sometimes wholly specified by this handle as a handle in this figure in the bottom holds NULL, Since it means that use of the tray was forbidden and the tray held NULL with as, it will be prevented that it is incorrect-surmised that the numerical value in the cell in the range of the value corresponding to a handle is a handle, and opening of a cell can be prevented from being delayed.

[0124]A tray can be used effectively, prohibition of use of the tray by a disable flag preventing delay of opening of the cell which is not canceled and used, when the tray which can be used is lost.

[0125]

[Effect of the Invention]The inside of two or more cell blocks which consist of a cell

which the garbage collection device concerning this invention is specified by the 1st sort address, and holds one data of numerical values other than the 2nd sort address and the 2nd sort address, Except for the cell block which can be followed from a predetermined route cell block by repeating the reference to other cell blocks via the tray which is specified by the 2nd sort address and holds the 1st sort address, So that two or more continuous cells which do not hold data in advance of the SUKABINJI processing which specifies an intact cell block, and said intact cell block and SUKABINJI processing which were specified may be arranged as one continuous field, A cell block is moved on the memory which consists of the 1st sort address and a cell, and compaction processing which updates the 1st sort address currently held at the tray is performed so that the relation of said reference may be maintained before and after movement of a cell block.

[0126]Whether in this garbage collection device, the cell contained in the 1st cell block holds the 2nd sort address. It is surmised that it does not mistake that the 2nd sort address is not held when the 2nd sort address is held, Specification of the intact cell block in said SUKABINJI processing is performed by presuming that the 2nd cell block containing the cell corresponding to the 1st sort address which the tray directed by the cell conjectured to hold the 2nd sort address holds has reference from the 1st cell block.

[0127]A tray (indirect addressing) is used for specification of a cell by this, In compaction processing, the 1st sort address currently held at the tray will be updated, the contents of the cell itself are not updated, and a compaction can be performed at high speed effectively (the garbage collector of the 3rd conventional example is a difference). These SUKABINJI processings and compaction processing are faced, (Like [ the 1st conventional example / of a garbage collector and the garbage collector of the 2nd conventional example ]) Since information is not exchanged with application using a reference flag etc., It is not necessary to secure the field for a reference flag, and becoming excessive load to execution of the application using a cell is prevented.

[0128]Let the tray which will direct the intact cell block newly specified by said SUKABINJI processing be an intact tray which does not hold the 1st sort address in an above garbage collection device. By this, in addition to the above effects, when use of a cell is newly required, the tray which finished use will be reused effectively.

[0129]In an above garbage collection device. The data which the cell holds is in the range corresponding to the 2nd sort address, And whenever the tray which will be directed when it considers that said data is the 2nd sort address is not an intact tray, if said data is the 2nd sort address, Said guess is performed, and when it considers that the numerical value newly substituted for a cell is the 2nd sort address, use of the tray which will be directed by said 2nd sort address shall be forbidden.

[0130]Thereby, in addition to an effect like point \*\*, use of the tray corresponding to the numerical value newly substituted for a cell is forbidden (since it is judged in the case of a guess whether the tray specified with the data substituted for a cell is an intact tray), Guessing accidentally that the numerical value in the same within the limits as the 2nd sort address is the 2nd sort address will be avoided.

[0131]In an above garbage collection device. When the tray of the predetermined number cannot be used by forbidding holding the 1st sort address or use and use of a new tray is required, all prohibitions of use to said tray shall be canceled. The tray which finished use in addition will be reused effectively, an incorrect guess being avoided by this in addition to an effect like point \*\*.

---

[Translation done.]